Perspectives in Science (2016) 8, 586—588

ScienceDirect
journal homepage: www.elsevier.com/pisc

Hybrid fault tolerant routing algorithm in NoC☆

Rimpy Bishnoi

Department of Information Technology, JB Institute of Engg. & Technology, Hyderabad, India

Received 21 February 2016; accepted 14 June 2016
Available online 18 July 2016

KEYWORDS
NoC;
SoC;
Routing

Summary Network-on-Chip has been a growing design paradigm with the rise in Multi-Processor System on Chip (MPSoCs) primarily due to its scalability. While regular meshes (2 or 3-dimensional) are the usual proposal for such a paradigm, a real chip may not follow it. Heterogeneous cores, hardware failures or manufacturing defects can possibly cause irregular topologies in a Network-on-Chip. Selection of a routing algorithm is an important challenge in NoC design as it affects power consumption, communication latency and overall system performance. Routing can be supported in such faulty environment by use of routing tables. But this is not a scalable solution as table size grows with network size. Logic Based Distributed Routing (LBDR) is proposed as a new routing implementation technique which offers compact routing implementation and fault tolerance without use of routing table. In this paper we propose a Hybrid Fault Tolerant Routing Algorithm (HFTRA), which aims to provide fault tolerance support in presence of on-chip link failures. Proposed routing is implemented with LBDR scheme. Analysis of the method has shown the effectiveness of proposed scheme as compared to routing tables when implemented using LBDR.

Introduction

The chip design concept has shifted from single core to multicore system design. As the number of cores on a single chip continues to increase, focus has shifted from computation to communication (Lee et al., 2008). Communication architecture greatly impacts the area, performance and energy consumption of overall multicore system design. To meet the challenges of current and future multicore architectures, a packet-switched interconnect Network-on-Chip (NoC) emerged as an alternative to the traditional bus based and point to point interconnects (Dally and Towles, 2001). NoC has proven to be a scalable, reliable and efficient communication paradigm (Benini et al., 2002).

Multicore systems consist of multiple heterogeneous Intellectual Property (IP) cores (CPU, memory controllers, DSP modules, etc.) that communicate through the underlying net-work infrastructure. As there could be many possible paths between a source and destination IP core, selection of some path may degrade the system's performance

☆ This article belongs to the special issue on Engineering and Material Sciences.

E-mail address: rimpybishnoi@gmail.com

http://dx.doi.org/10.1016/j.pisc.2016.06.028

drastically (Guerrier and Greiner, 2000). Therefore the way messages are routed is one of key challenging problem in NoC design (Marculescu et al., 2009). Selection of a given routing algorithm is used to determine the way messages are routed through the network. Routing algorithm affects the communication latency, power consumption and overall performance of underlying network architecture (Bjerregaard and Mahadevan, 2006).

Based on the way next hop along the route is determined, routing algorithm can be deterministic or adaptive (Rantala et al., 2006). Deterministic routing algorithm does not take into account the current network status and always generates a fixed path for each source destination pair in network. On the other side, adaptive routing generates multiple paths for each source destination pair and based on current network status it selects the final path. Based on the number of hops a message takes, routing algorithms can be minimal or non-minimal. Performance of routing algorithm also depends on its implementation. Routing algorithm can be implemented as source or distributed routing (Rantala et al., 2006). In source routing, complete path from source to destination is computed at source node and stored in packet header. In distributed routing, path is computed at each hop based on destination address stored in packet header. To address the challenges posed by irregular topologies, both techniques make use of routing tables. This is not an efficient solution as size of table increases when network size increases resulting in increase of size of router and also its complexity. Recently, Logic Based Distributed Routing (LBDR) (Flich and Duato, 2008) has been proposed as a new routing implementation technique which is compact and offers high fault tolerance. With increasing chip densities, the probability that a manufacturing defect affects communication system is also increased. Defects affecting links of NoC can be handled by fault tolerant routing algorithms.

In this paper we propose a novel Hybrid Fault Tolerant Routing Algorithm (HFTRA) that uses minimal and non-minimal paths to route the traffic. Basically, it uses different virtual channels, each implementing a different routing algorithm. All the algorithms are implemented using LBDR which offers great advantage when implemented using routing tables. In absence of failures, HFTRA behaves like minimal path routing for a given source destination node. When fault occurs, it searches for minimal path. If unavailable, it routes the traffic to virtual channels having non-minimal routing. In this way it enjoys the benefit of both minimal and non-minimal routings by adapting to fault-tolerance and bypassing defected links where necessary otherwise stick to minimal for efficiency.

The rest of the paper is structured as follows. In ''Related work'' section an overview of fault tolerance is presented. ''LBDR overview'' section gives a brief overview of LBDR technique. ''Proposed hybrid fault tolerant routing'' section is devoted to the explanation of proposed method. ''Analysis'' section analyzes the proposed scheme. Finally, ''Conclusion and future work'' section concludes and directions for future work are introduced.

Related work

Duato et al. (1997) explained the effect of faults on the correctness of routing algorithm. They presented common fault models along with various definitions related to fault tolerant routing algorithms. Sui and Wang (1997) proposed a constrained fault model in which faults form a convex shape (ring or chain) and route message alongside faulty regions until it is back in its normal route. In their approach, flits use specific virtual channels based on what route they are on. This resulted in bad utilization of all available virtual channels. An improvement to this approach proposed by Rezazadeh et al. (2009) uses lesser virtual channels but allows U-turns (180 degrees) to support non-minimality of the route. Li et al. (2009) applies a multi-level congestion control for load balancing and detours packet based on the same in case a fault is encountered. Certain approaches use deactivation of healthy nodes over virtual channels to ensure deadlock freedom.

LBDR overview

Distributed routing algorithms often deploy turn restrictions to obtain freedom from deadlock and livelock. A routing algorithm can therefore be solely depicted by its routing (turn) restrictions. Distributed routing algorithms can be implemented as source and destination routing with or without using routing table. To map irregular topology, both schemes require routing tables. As the size of routing table grows with network size, tables are not scalable solution for representing irregular topology. Logic Based Distributed Routing (LBDR) is proposed as a new mechanism for routing implementation without using routing table (Flich and Duato, 2008). Any topology derived from initial 2D mesh can be easily mapped using LBDR. LBDR uses 2 sets of configuration bits to map topology and the routing algorithm to the NoC. In each output direction of a switch, a routing bit Rxy indicates whether a turn towards '*y*' on the next hop in '*x*' direction is allowed or not and the connectivity bit Cz indicates the connectivity (which may not exist due to irregularity in the topology or faults) with the switch in the *z* direction. For a 2-dimensional mesh, there would be 2 routing bits and 1 connectivity bit per output direction, hence a total of 12 routing and 4 connectivity bit per switch. Table 1 shows the routing and connectivity bits corresponding to XY routing in 3×3 topology. XY routing restricts packets to take YX turn. Logic of distributed routing computation is as described below:

A direction (say North (N)) can be taken if one of the following condition is satisfied:

1) Packet destination is in North (N) direction.
2) Packet destination is in NE direction and an East (E) turn is allowed on the next hop.
3) Packet destination is in NW direction and a West (W) turn is allowed on the next hop.

Similarly, LBDR computes availability of all other directions and one of the available directions is chosen.

Proposed hybrid fault tolerant routing

In this section we propose Hybrid Fault Tolerant Routing Algorithm (HFTRA). Proposed algorithm is Hybrid in sense that it suggests an effective way to combine multiple routing

Table 1 Routing and connectivity bits for XY routing.

ID	Rne	Rnw	Ren	Res	Rse	Rsw	Rwn	Rws	Cn	Ce	Cw	Cs
0	1	1	1	1	0	1	1	1	0	1	0	1
1	1	1	1	1	0	0	1	1	0	1	1	1
2	1	1	1	1	1	0	1	1	0	0	1	1
3	0	1	1	1	0	1	1	1	1	1	0	1
4	0	0	1	1	0	0	1	1	1	1	1	1
5	1	0	1	1	1	0	1	1	1	0	1	1
6	0	1	1	1	1	1	1	1	1	1	0	0
7	0	0	1	1	1	1	1	1	1	1	1	0
8	1	0	1	1	1	1	1	1	1	0	1	0

algorithms for different virtual channels, each one implementing a separate routing algorithm.

In presence of single or multiple failures, first it tries to route the traffic along minimal path only. When all available minimal paths become faulty then it route the traffic along the non- minimal path. On the event of a failure, it looks for the virtual channel which can bypass faulty routes. For example, if there are initially three virtual channels (VC1, VC2, VC3) implementing XY, YX and west first routing. If the failure is in east direction of the source, then the proposed algorithm forwards the packet using VC3 as it offers fault free minimal or non-minimal paths. In the absence of failures, proposed algorithm can route traffic to any of the virtual channels, thus improving the overall performance. Moreover, using different algorithms on the VCs tries to distribute traffic evenly to all the links. Also, all algorithms are implemented using LBDR, which further improves the performance as it does not require any access to routing table. HFTRA maintains the deadlock freedom by not allowing any transition between virtual channels.

Analysis

In this section we analyse the impact of HFTRA implemented using LBDR as compared to routing tables. Idea is that when we use LBDR to implement the HFTRA, irrespective of the number of virtual channels, it uses a single routing logic. It captures the rules of different routing algorithms by using a separate set of LBDR bits corresponding to each routing algorithm. On the other side, routing table implementation requires number of routing tables equal to the number of virtual channels, each one storing routes calculated according to different routing algorithms. Hence this increases the overall area overhead as compared to proposed one.

Conclusion and future work

In this paper we have presented a hybrid fault tolerant routing algorithm in NoC. Proposed method is based on the fact that if we use different algorithms for different virtual channels then it would offer fault tolerance by choosing the virtual channel which offers fault free path. Area overhead can be controlled by implementing the algorithms using LBDR. In future, we will see the impact on fault tolerance by allowing the transition between virtual channels while preserving deadlock freedom.

References

Benini, L., De, G., Micheli, 2002. Networks on chips: a new soc paradigm. Computer 35 (1), 70—78.

Bjerregaard, T., Mahadevan, S., 2006. A survey of research and practices of network-on-chip. ACM Comput. Surv. 38 (June (1)).

Dally, W.J., Towles, B., 2001. Route packets, not wires: on-chip interconnection networks. In: Design Automation Conference, 2001. Proceedings, pp. 684—689.

Duato, J., Yalamanchili, S., Ni, L.M., 1997. Interconnection Networks — An Engineering Approach. IEEE.

Flich, J., Duato, J., 2008. Logic-based distributed routing for NoCs. CAL 7 (1), 13—16.

Guerrier, P., Greiner, A., 2000. A generic architecture for on-chip packet- switched interconnections. In: Design, Automation and Test in Europe Conference and Exhibition 2000. Proceedings, pp. 250—256.

Lee, H.G., et al., 2008. On-chip communication architecture exploration: a quantitative evaluation of point-to-point, bus, and network-on-chip approaches. ACM Trans. Des. Autom. Electron. Syst. 12 (May (3)), 231—2320.

Li, X., et al., 2009. Fault-tolerant routing algorithm for network-on-chip based on dynamic xy routing. Wuhan Univ. J. Nat. Sci. 14 (4), 343—348.

Marculescu, R., et al., 2009. Outstanding research problems in NoC design: system, microarchitecture, and circuit perspectives. IEEE Trans. CAD-ICS 28 (1), 3—21.

Rantala, V., et al., 2006. Network on Chip Routing Algorithms.

Rezazadeh, A., Fathy, M., Rahnavard, G., 2009. An enhanced fault-tolerant routing algorithm for mesh network-on-chip. In: International Conference on Embedded Software and Systems, 2009. ICESS'09, pp. 505—510.

Sui, P.-H., Wang, S.-D., 1997. An improved algorithm for fault-tolerant wormhole routing in meshes. IEEE Trans. Comput. 46 (9), 1040—1042.